Remarks

The present amendment responds to the Official Action dated September 28, 2004. The Official Action objected to claims 11, 15, 18, and 36 as informal. The Official Action rejected claims 1-18 under 35 U.S.C. §101 as purportedly directed to non-statutory subject matter. Claims 1, 10-15, 19, and 28-33 were rejected under 35 U.S.C. §103(a) based on Faraboschi et al. U.S. Patent No. 5,930,508 ("Faraboschi") in view of Roediger et al. U.S. Patent No. 6,305,014 ("Roediger"). Claims 2-9 and 20-27 were rejected under 35 U.S.C. §103(a) based on Faraboschi and Roediger in view of McKinsey et al. U.S. Patent No. 6,675,380 ("McKinsey"). Claims 18 and 36 were rejected under 35 U.S.C. §103(a) based on McKinsey. Claims 16-17 and 34-35 were rejected under 35 U.S.C. §102(e) as being anticipated by McKinsey. These grounds of rejection are addressed below.

Claims 1, 11, 15, 18, 19, and 36 have been amended to be more clear and distinct. More particularly, claims 1 and 19 have been amended to be more clear and distinct by clarifying that "at least some of the plurality of VLIW instructions" are allocated to VIM based on their lifetime. Claim 1 has been further amended to address the 35 U.S.C. §101 rejection as described in further detail below. Claims 11, 15, 18, and 36 have been amended to address the informality objections as discussed below. Claims 1-36 are presently pending.

Informality Objections to Claims 11, 15, 18, and 36

Claims 11 and 15 have been amended to expand the acronyms "LV" and "XV." Support for these expansions can be found, for example, at page 8, line 3 of the Specification.

Claims 18 and 36 have been amended to replace an improper "." with a ";".

35 U.S.C. Sec. 101 Rejection of Claims 1-18

The Official Action at page 2, item 7 suggests that the present claims are not statutory subject matter because they "recite components of allocating memory for VLIW, representing functional descriptive material without a computer readable medium or computer implemented, program/method per se are not tangibly embodied." Applicants do not acquiesce in this analysis. Furthermore, Applicants believe this rejection is unclear since claims 1-18 are directed to a method which is clearly patentable subject matter.

However, the term "source program" is used in claim 1. In this regard, MPEP §2106 IV(A) states

> Computer programs are often recited as part of a claim. Office personnel should determine whether the computer program is being claimed as part of an otherwise statutory manufacture or machine. In such a case, the claim remains statutory irrespective of the fact that a computer program is included in the claim. The same result occurs when a computer program is used in a computerized process where the computer executes the instructions set forth in the computer program. Only when the claimed invention taken as a whole is directed to a mere program listing, i.e., to only its description or expression, is it descriptive material *per se* and hence nonstatutory. (emphasis added)

The MPEP clearly sets forth the test for nonstatutory subject matter. Claims 1-18 are clearly written in method form, a proper statutory class of invention. Although the term "source program" is included in claim 1, the claimed invention as a whole is not directed to a mere program listing. Consequently, claims 1-18 clearly satisfies the test set forth for nonstatutory subject matter in the MPEP.

The Art Rejections

As addressed in greater detail below, Faraboschi, Roediger, and McKinsey do not support the Official Action's reading of them and the rejections based thereupon should be reconsidered and withdrawn. Further, the Applicant does not acquiesce in the analysis of Faraboschi, Roediger, and McKinsey made by the Official Action and respectfully traverses the Official Action's analysis underlying its rejections.

Claims 1, 10-15, 19, and 28-33 were rejected under 35 U.S.C. §103(a) based on Faraboschi in view of Roediger. Faraboschi describes compacting very long instruction words (VLIW) in a processor by eliminating no operation (NOP) codes from the VLIW instructions. Faraboschi, col. 2, lines 66-67. The NOP codes in a VLIW instruction correspond to operation codes for functional units that are not needed to execute the VLIW instruction. Faraboschi, col. 1, lines 52-57. To this end, Faraboschi's approach involves identifying each word of an instruction that does not contain a NOP code, generating a dispersal code for each identified word, generating a delimiter code for each identified word and storing each identified word along with the corresponding dispersal and delimiter code. Faraboschi, col. 3, lines 1-10. Unlike the present invention, Faraboschi addresses a totally different problem of compacting VLIW instructions by eliminating NOP codes from an instruction. By contrast, the present invention addresses techniques for allocating VLIW instructions to VLIW instruction memory (VIM). See page 2, lines 8-10 of the Specification, for example, for a discussion of VIM allocation.

Claim 1, as presently amended, recites "allocating at least some of the plurality of VLIW instructions to VIM." See also claim 19. With respect to claim 15, the Official Action attempts

to equate eliminating NOP codes from a VLIW as taught by Faraboshchi with reducing redundant loads of very long instruction word instruction memory. Applicants respectfully disagree. NOP codes are used under the assumption that each execution unit should perform an operation on every processor cycle. Faraboshchi, col. 1, lines 41-43. Rather than eliminating the storage of NOP codes in memory, claim 15, as presently amended, reduces redundant loading of a load VLIW (LV) instruction by "selecting a load VLIW (LV) instruction in a current node; and placing the LV instruction in a new node which is closer to a program start node if an execution frequency of the new node is lower than an execution frequency of the current node, and if a maximum number of VIM lines is not exceeded." Faraboshchi's disclosure is silent with respect to selecting and placing of an LV instruction as presently claimed. See also claim 33. Furthermore, the Official Action admits that Faraboschi does not explicitly disclose determining a lifetime of each of said plurality of VLIW instructions and relies on Roediger accordingly.

Roediger does not cure the deficiencies of Faraboschi. Roediger addresses an instruction scheduler in an optimizing compiler by determining the lifetimes of fixed registers. Roediger, col. 1, lines 62-65. During the compilation, symbolic registers are mapped to fixed registers based on the lifetime of the fixed registers. However, Roediger does not address "allocating at least some of the plurality of VLIW instructions to VIM based on the lifetime of said plurality of VLIW instructions" as presently claimed in claim 1.

The Official Action suggests that Roediger and Faraboschi should be combined "to provide faster execution of [a] computer program." Applicants travers any such basis for combining these two items. Further, Roediger and Faraboschi cannot simply be combined to

meet the presently claimed invention. By way of example, if the teachings of Roediger and Faraboschi were combined, how would combining the compression of VLIW instructions containing NOP codes as taught by Faraboschi with the determination of the lifetime of fixed registers as taught by Roediger specify how to allocate VLIW instructions to VIM? Referring to the Specification at page 2, lines 17-21, unlike the Faraboschi and Roediger combination, the present invention advantageously addresses an approach in which a small VIM size may be used even when a software application demands a number of VLIWs larger than can be fit into the physical VIM size of a processor.

Claims 2-9 and 20-27 were rejected under 35 U.S.C. §103(a) based on Faraboschi and Roediger in view of McKinsey. McKinsey fails to cure the deficiencies of Faraboschi and Roediger. Since claims 2-9 depend from and contain all the limitations of claim 1, as presently amended, claims 2-9 distinguish from the references in the same manner as claim 1. Since claims 20-27 depend from and contain all the limitations of claim 19, as presently amended, claims 20-27 distinguish from the references in the same manner as claim 19.

Furthermore, McKinsey addresses a path speculating instruction scheduler. Mckinsey, Abstract. With McKinsey's system instructions may be executed out of order to improve the performance of a processor. In so doing, McKinsey's system includes creating control flow graphs made up of blocks of instruction. The control flow graphs are utilized to determine whether a block of instructions can be executed in the same machine cycle, or may be executed in different machine cycles depending on the available resources in the processor and data

dependencies between instructions. McKinsey, col. 5, lines 29-42. Unlike the present invention, McKinsey's utilization of control graphs does not address allocation of a VLIWs to VIM.

Unlike McKinsey, the present invention utilizes a control graph to determine the lifetime of a VLIW. Referring to page 10, lines 3-8 of the Specification, the term "lifetime of a VLIW" is defined to extend from a node in a control graph containing the LV instruction which loads the VLIW into a VIM line until the node that uses the contents of the same VIM line with an XV instruction for the last time in the program execution. It should be noted that lifetime of a VLIW is quite different than the lifetime of register. The term "lifetime" when applied to register allocation typically refers to a "program relative measure of time that a variable is actively used requiring its storage remain in a physical register of the processor." Specification, page 9, lines 227-29. Claim 2, as presently amended, clarifies the step of determining the lifetime of a VLIW when it recites "determining a control flow graph for the input source program containing said plurality of VLIW instructions; determining a VLIW flow graph for said plurality of VLIW instructions; and determining VLIW interference." McKinsey does not disclose and does not make obvious determining the lifetime of VLIW by "determining a control flow graph for the input source program containing said plurality of VLIW instructions; determining a VLIW flow graph for said plurality of VLIW instructions; and determining VLIW interference," as claimed in claim 2.

Claims 18 and 36 were rejected under 35 U.S.C. §103(a) based on McKinsey. This rejection is unclear in that there is only one reference without a reliance on common knowledge,

or inherency upon which the §103 rejection is made. Unlike the present invention, McKinsey does not address a VLIW environment as described above.

The Official Action suggests that McKinsey purportedly discloses determining an interference graph comprising VLIW nodes. However, the only place "VLIW" is referenced in McKinsey is in the "Other Publications" section. McKinsey does not teach and does not suggest "determining an interference graph comprising VLIW nodes in which every VLIW node of the interference graph corresponds to one VLIW instruction; inserting an undirected edge into the interference graph between two VLIW nodes if the two VLIW instructions belong to a live-out set of the same node of the VLIW flow graph; and coloring the VLIW graph nodes such that adjacent VLIW nodes are colored in different colors and each color corresponds to a different VIM line," as presently claimed in claim 18. See also claim 36.

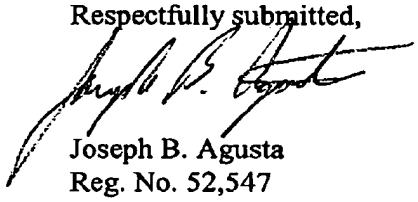
Claims 16-17 and 34-35 were rejected under 35 U.S.C. §102(e) as being anticipated by McKinsey. As stated above, McKinsey does not address a VLIW environment. Consequently, McKinsey does not disclose and does not make obvious "determining a VLIW flow graph by solving VLIW flow equations," as claimed in claim 16. Furthermore, McKinsey does not disclose and does not make obvious a "means for determining a VLIW flow graph by solving VLIW flow equations," as claimed in claim 34.

The relied upon references fail to recognize and address the problem of allocating VLIWs to VIM in the manner advantageously addressed by the present claims. The claims as presently amended are not taught, are not inherent, and are not obvious in light of the art relied upon.

Conclusion

All of the presently pending claims, as amended, appearing to define over the applied references, withdrawal of the present rejection and prompt allowance are requested.

Respectfully submitted,



Joseph B. Agusta
Reg. No. 52,547
Priest & Goldstein, PLLC
5015 Southpark Drive, Suite 230
Durham, NC 27713-7736
(919) 806-1600